

ThreatQuotient

A Securonix Company



Feedly CDF

Version 2.1.3

August 02, 2026

ThreatQuotient

20130 Lakeview Center Plaza Suite 400
Ashburn, VA 20147



ThreatQ Supported

Support

Email: tq-support@securonix.com

Web: <https://ts.securonix.com>

Phone: 703.574.9893

Contents

Warning and Disclaimer	3
Support	4
Integration Details	5
Introduction	6
Prerequisites	7
Installation	8
Configuration	9
Feedly Parameters.....	9
Feedly Threat Intelligence Parameters	11
ThreatQ Mapping	14
Feedly.....	14
Feedly Threat Intelligence	23
Average Feed Run	28
Feedly.....	28
Feedly Threat Intelligence	29
Known Issues / Limitations	30
Change Log	31

Warning and Disclaimer

ThreatQuotient, Inc. provides this document "as is", without representation or warranty of any kind, express or implied, including without limitation any warranty concerning the accuracy, adequacy, or completeness of such information contained herein.

ThreatQuotient, Inc. does not assume responsibility for the use or inability to use the software product as a result of providing this information.

Copyright © 2026 ThreatQuotient, Inc.

All rights reserved. This document and the software product it describes are licensed for use under a software license agreement. Reproduction or printing of this document is permitted in accordance with the license agreement.

Support

This integration is designated as **ThreatQ Supported**.

Support Email: tq-support@securonix.com

Support Web: <https://ts.securonix.com>

Support Phone: 703.574.9893

Integrations/apps/add-ons designated as **ThreatQ Supported** are fully supported by ThreatQuotient's Customer Support team.

ThreatQuotient strives to ensure all ThreatQ Supported integrations will work with the current version of ThreatQuotient software at the time of initial publishing. This applies for both Hosted instance and Non-Hosted instance customers.



ThreatQuotient does not provide support or maintenance for integrations, apps, or add-ons published by any party other than ThreatQuotient, including third-party developers.

Integration Details

ThreatQuotient provides the following details for this integration:

Current Integration Version	2.1.3
Compatible with ThreatQ Versions	>= 5.6.0
Support Tier	ThreatQ Supported

Introduction

The Feedly integration for ThreatQ allows a user to ingest feeds as reports and related objects from Team feeds on Feedly.

You cannot pull from a Feedly Personal feed.

The CDF includes the following feeds:

- **Feedly** - ingests reports and other related objects from Feedly.
- **Feedly Threat Intelligence** - ingests STIX 2.1 Reports from a Feedly stream and can optionally include related Indicators, Malware, Adversaries, and MITRE ATT&CK techniques based on the configured settings.

The integration ingests the following system object types:

- | | |
|---|--|
| • Adversaries <ul style="list-style-type: none">◦ Adversary Attributes | • Malware <ul style="list-style-type: none">◦ Malware Attributes |
| • Attack Patterns <ul style="list-style-type: none">◦ Attack Pattern Attributes | • Reports <ul style="list-style-type: none">◦ Report Attributes |
| • Campaigns <ul style="list-style-type: none">◦ Campaign Attributes | • Signatures <ul style="list-style-type: none">◦ Signature Attributes |
| • Courses of Action <ul style="list-style-type: none">◦ Course of Action Attributes | • Tools <ul style="list-style-type: none">◦ Tool Attributes |
| • Identities <ul style="list-style-type: none">◦ Identity Attributes | • Vulnerabilities <ul style="list-style-type: none">◦ Vulnerability Attributes |
| • Indicators <ul style="list-style-type: none">◦ Indicator Attributes | • TTPs <ul style="list-style-type: none">◦ TTP Attributes |
| • Intrusion Sets <ul style="list-style-type: none">◦ Intrusion Set Attributes | |

Prerequisites

The following is required to run the integration:

- **All feeds** - a valid Feedly API Token and Feedly API Steam ID is required.
- **Feedly Threat Intelligence** - the Feedly Threat Intelligence feed requires your Feedly account to have an Enterprise license.

Installation

Perform the following steps to install the integration:



The same steps can be used to upgrade the integration to a new version.

1. Log into <https://marketplace.threatq.com/>.
2. Locate and download the integration yaml file.
3. Navigate to the integrations management page on your ThreatQ instance.
4. Click on the **Add New Integration** button.
5. Upload the yaml file using one of the following methods:
 - Drag and drop the file into the dialog box
 - Select **Click to Browse** to locate the file on your local machine



ThreatQ will inform you if the feed already exists on the platform and will require user confirmation before proceeding. ThreatQ will also inform you if the new version of the feed contains changes to the user configuration. The new user configurations will overwrite the existing ones for the feed and will require user confirmation before proceeding.

6. If prompted, select the individual feeds to install and click **Install**. The feed will be added to the integrations page.

You will still need to [configure and then enable](#) the feed.

Configuration



ThreatQuotient does not issue API keys for third-party vendors. Contact the specific vendor to obtain API keys and other integration-related credentials.

To configure the integration:

1. Navigate to your integrations management page in ThreatQ.
2. Select the **Commercial** option from the *Category* dropdown (optional).



If you are installing the integration for the first time, it will be located under the **Disabled** tab.

3. Click on the integration entry to open its details page.
4. Enter the following parameters under the **Configuration** tab:

Feedly Parameters

PARAMETER	DESCRIPTION
Feedly API Token	Your Feedly API Token.
Feedly API Stream ID	Your Feedly Stream ID.
Ingest Keywords As	Select the type of object Keywords should be ingested as. Options include: ◦ Tags (<i>default</i>) ◦ Attributes
Parsed IOC Types	Select the IOC types to automatically parse from the content. Options include: ◦ CVE (<i>default</i>) ◦ SHA-512 (<i>default</i>)

PARAMETER

DESCRIPTION

- | | |
|------------------------------------|---------------------------------------|
| ◦ IP Address
(<i>default</i>) | ◦ Email Address
(<i>default</i>) |
| ◦ IPv6 Address | ◦ Registry Key |
| ◦ CIDR Block | ◦ URL (<i>default</i>) |
| ◦ MD5 (<i>default</i>) | ◦ FQDN (<i>default</i>) |
| ◦ SHA-1 (<i>default</i>) | ◦ File Path |
| ◦ SHA-256
(<i>default</i>) | ◦ Filename |
| ◦ SHA-384 | |



Not all IOC types may be included in the Feedly API.

Ingest CVEs As:

Select the type of objects CVEs should be ingested as. Options include:

- Indicators
- Vulnerabilities (*default*)

Ingest MITRE ATT&CK As

Select how to ingest MITRE ATT&CK techniques as in ThreatQ. Options include:

- Attack Pattern (*default*)
- TTP

MITRE ATT&CK Naming Format

Select the naming format used for MITRE ATT&CK techniques ingested from all Feedly feeds. Options include:

- TID - Technique Name (*default*)
- Technique Name (Enterprise TID)

PARAMETER

DESCRIPTION

Enable SSL Certificate Verification

Enable this parameter if the feed should validate the host-provided SSL certificate.

Disable Proxies

Enable this parameter if the feed should not honor proxies set in the ThreatQ UI.

< Feedly



Disabled ☒ Enabled

Uninstall

Additional Information

Integration Type: Feed

Version:

Configuration Activity Log

Authentication

Feedly API Token

Enter your Feedly API Token found under "Manage Teams -> API"

Feed Configuration

Feed Stream IDs can be found by accessing the "Settings" for your feed, and navigating to the "Sharing" tab.

Feedly API Stream ID

Enter a Feedly Stream ID associated with your curated feed.

Ingestion Options

Ingest Keywords As

Choose whether you want keywords ingested as Tags, attributes, or both

☒ Tags

☐ Attributes

Parsed IOC Types

Select the IOC types you would like to automatically parse from the content. Not all IOC types may be included in the Feedly API.

☒ CVE

☒ IP Address

☐ IPv6 Address

☐ CIDR Block

☒ MD5

☒ SHA-1

Feedly Threat Intelligence Parameters

PARAMETER

DESCRIPTION

Feedly API Token

Your Feedly API Token.

PARAMETER	DESCRIPTION
Feedly API Stream ID	Your Feedly Stream ID.
Do not Ingest Description	Enable this parameter to exclude report descriptions from ingestion. This is particularly useful when the Feedly Threat Intelligence feed and the base Feedly feed share the same Feedly Stream ID, as it prevents duplicate report descriptions from being created. With this parameter enabled, the description will be sourced solely from the base Feedly feed.
Parsed IOC Types	Select the IOC types to automatically parse from the content. Options include: ◦ CVE (<i>default</i>) ◦ IP Address (<i>default</i>) ◦ IPv6 Address ◦ CIDR Block ◦ MD5 (<i>default</i>) ◦ SHA-1 (<i>default</i>) ◦ SHA-256 (<i>default</i>) ◦ SHA-384 ◦ SHA-512 (<i>default</i>) ◦ Email Address (<i>default</i>) ◦ Registry Key ◦ URL (<i>default</i>) ◦ FQDN (<i>default</i>) ◦ File Path ◦ Filename  Not all IOC types may be included in the Feedly API.
Ingest CVEs As	Select the type of objects CVEs should be ingested as. Options include: ◦ Indicators ◦ Vulnerabilities (<i>default</i>)

PARAMETER	DESCRIPTION
Ingest MITRE ATT&CK As	Select how to ingest MITRE ATT&CK techniques as in ThreatQ. Options include: ◦ Attack Pattern (<i>default</i>) ◦ TTP
MITRE ATT&CK Naming Format	Select the naming format used for MITRE ATT&CK techniques ingested from all Feedly feeds. Options include: ◦ TID - Technique Name (<i>default</i>) ◦ Technique Name (Enterprise TID)
Ingest STIX Indicator Patterns as Signatures	Enable this option will result in STIX indicators to be ingested as signatures. If disabled, indicator values will be ingested as indicators.
Enable SSL Certificate Verification	Enable this parameter if the feed should validate the host-provided SSL certificate.
Disable Proxies	Enable this parameter if the feed should not honor proxies set in the ThreatQ UI.

5. Review any additional settings, make any changes if needed, and click on **Save**.
6. Click on the toggle switch, located above the *Additional Information* section, to enable it.

ThreatQ Mapping

Feedly

The Feedly feed ingests articles from a Feedly stream, as Report objects, along with any related context (i.e. tags, indicators, malware, etc.).

GET <https://cloud.feedly.com/v3/streams/contents>

Sample Response:

```
{
  "continuation": "17eb965fd82:8f60c7:26e2bd2e",
  "id": "enterprise/threatquotient/category/4b1e06a8-b4de-4e74-9593-ac879d1d3d23",
  "items": [
    {
      "alternate": [
        {
          "href": "https://packetstormsecurity.com/files/165814/wpcfct102-xssaccess.txt",
          "type": "text/html"
        }
      ],
      "canonicalUrl": "https://packetstormsecurity.com/files/165814/wpcfct102-xssaccess.txt",
      "categories": [
        {
          "id": "enterprise/threatquotient/category/4b1e06a8-b4de-4e74-9593-ac879d1d3d23",
          "label": "Threat Intel"
        }
      ],
      "commonTopics": [
        {
          "id": "nlp/f/topic/2440",
          "label": "Vulnerabilities",
          "salienceLevel": "about",
          "score": 1.0,
          "type": "topic"
        }
      ],
    }
  ]
}
```

```

        "id": "nlp/f/topic/3003",
        "label": "Cyber Security",
        "salienceLevel": "about",
        "score": 1.0,
        "type": "topic"
    },
    ],
    "crawled": 1643826412301,
    "entities": [
        {
            "id": "vulnerability/m/entity/CVE-2021-24247",
            "label": "CVE-2021-24247",
            "mentions": [
                {
                    "text": "CVE-2021-24247"
                }
            ],
            "vulnerabilityInfo": {
                "cvssScore": 5.4,
                "description": "The Contact Form Check Tester
WordPress plugin through 1.0.2 settings are visible to all registered
users in the dashboard and are lacking any sanitisation. As a result,
any registered user, such as subscriber, can leave an XSS payload in
the plugin settings, which will be triggered by any user visiting
them, and could allow for privilege escalation. The vendor decided to
close the plugin.",
                "hasExploit": true,
                "hasPatch": false
            }
        },
        {
            "type": "mitreAttack",
            "disambiguated": true,
            "id": "nlp/f/entity/gz:mi:x-mitre-tactic-
ffd5bcee-6e16-4dd2-8eca-7b3beedf33ca",
            "label": "Initial Access (Enterprise TA0001)",
            "mentions": [],
            "salienceLevel": "mention",
            "causes": [
                {
                    "id": "nlp/f/entity/gz:mi:attack-pattern-
a62a8db3-f23a-4d8f-afd6-9dbc77e7813b",

```

```

        "label": "Phishing (Enterprise T1566)"
      }
    ]
  },
  "estimatedCVSS": {
    "category": "HIGH"
  },
  "fingerprint": "b5dfd074",
  "fullContent": "<!DOCTYPE html PUBLIC \"-//W3C//DTD HTML 4.0
Transitional//EN\" \"http://www.w3.org/TR/REC-html40/
loose.dtd\">\n<html><body><div><div><div id=\"m\">\n  \n\n  \n
\n  \n  \n  <p></p><h1>WordPress Contact Form Check Tester
1.0.2 XSS / Access Control</h1>\n<dl id=\"F165814\"><dt><a
href=\"https://packetstormsecurity.com/files/download/165814/
wpcfct102-xssaccess.txt\" title=\"Size: 0.7 KB\"><strong>WordPress
Contact Form Check Tester 1.0.2 XSS / Access Control</strong></a></
dt>\n<dd>Posted <a href=\"https://packetstormsecurity.com/files/date/
2022-02-02/\" title=\"16:49:09 UTC\">Feb  2, 2022</a></
dd>\n<dd>Authored by <a href=\"https://packetstormsecurity.com/files/
author/13412/\">0xB9</a></dd>\n<dd><p>WordPress Contact Form Check
Tester plugin version 1.0.2 suffers from broken access control and
cross site scripting vulnerabilities.</p></dd>\n<dd><span>tags</span>
| <a href=\"https://packetstormsecurity.com/files/tags/
exploit\">exploit</a>, <a href=\"https://packetstormsecurity.com/
files/tags/vulnerability\">vulnerability</a>, <a href=\"https://
packetstormsecurity.com/files/tags/xss\">xss</a></
dd>\n<dd><span>advisories</span> | <a href=\"https://
packetstormsecurity.com/files/cve/CVE-2021-24247\">CVE-2021-24247</
a></dd>\n<dd><span>MD5</span> |
<code>6571a974217db95c175bd945e2b0d575</code></dd>\n<dd><a
href=\"https://packetstormsecurity.com/files/download/165814/
wpcfct102-xssaccess.txt\" title=\"Size: 0.7 KB\"
rel=\"nofollow\">Download</a> | <a href=\"https://
packetstormsecurity.com/files/favorite/165814/\"
rel=\"nofollow\">Favorite</a> | <a href=\"https://
packetstormsecurity.com/files/165814/WordPress-Contact-Form-Check-
Tester-1.0.2-XSS-Access-Control.html\">View</a></dd>\n</dl><p></
p><h1>WordPress Contact Form Check Tester 1.0.2 XSS / Access
Control</h1>\n<div>\n\n<pre><code># Exploit Title: WordPress Plugin
Contact Form Check Tester 1.0.2 - Broken Access Control<br># Date:
2/28/2021<br># Author: 0xB9<br># Software Link: https://

```

```
wordpress.org/plugins/contact-fo...ck-tester/<br># Version:  
1.0.2<br># Tested on: Windows 10<br># CVE: CVE-2021-24247<br><br>1.  
Description:<br>The plugin settings are visible to all registered  
users in the dashboard.<br>A registered user can leave a payload in  
the plugin settings.<br><br>2. Proof of Concept:<br>- Register an  
account<br>- Navigate to the dashboard<br>- Go to CF7 Check Tester  
& Settings<br>- Add a form<br>- Add a field to the form<br>- Put  
in a payload in either Field selector or Field value  
\\>&lt;script&gt;alert(1)&lt;/script&gt;<br>- Save<br>Anyone who  
visits the settings page will execute the payload.<br><br></code></  
pre>\n</div>\n \n      \n      \n      </div>\n      \n      </div></div></  
body></html>",  
    "id": "4/bnDLbGIuwgfYUHoNIoHFzwwucR/Wg3zkOf7t/  
Xc0Q=_17ebbb07b0d:f47321:aa31659c",  
    "language": "en",  
    "leoSummary": {  
        "sentences": [  
            {  
                "text": "Palo Alto Networks customers are  
protected against the types of BEC threats discussed in this blog by  
products including Cortex XDR and the WildFire, Threat Prevention,  
AutoFocus and Advanced URL Filtering subscription services for the  
Next-Generation Firewall .",  
                "position": 16,  
                "score": 0.316  
            }  
        ]  
    },  
    "origin": {  
        "htmlUrl": "https://packetstormsecurity.com/",  
        "streamId": "feed/http://packetstormsecurity.org/  
exploits.xml",  
        "title": "Exploit Files \u2248 Packet Storm"  
    },  
    "originId": "https://packetstormsecurity.com/files/165814/  
wpcfct102-xssaccess.txt",  
    "published": 1643820549000,  
    "sources": [  
        {  
            "feedlyFeedType": "WebAlert",  
            "searchTerms": {  
                "isComplexFilter": false,  
                "parts": [  

```

```

        {
            "id": "nlp/f/publicationBucket/
byf:cybersecurity-bundle",
            "label": "Cybersecurity"
        },
        {
            "text": "HIGH"
        },
        {
            "id": "nlp/f/entity/wd:13166",
            "label": "WordPress"
        }
    ]
},
    "streamId": "feed/https://feedly.com/f/alert/
704a6215-d181-427e-b1a4-d50032e51968",
    "title": "Wordpress Vulns"
}
],
"memes": "Website",
"summary": {
    "content": "WordPress Contact Form Check Tester plugin
version 1.0.2 suffers from broken access control and cross site
scripting vulnerabilities.",
    "direction": "ltr"
},
"title": "WordPress Contact Form Check Tester 1.0.2 XSS /
Access Control",
"keywords": [
    "My Software",
    "Update"
],
"unread": true,
"visual": {
    "url": "none"
},
"indicatorsOfCompromise": {
"exports": [
    {
        "type": "markdown",
        "url": "https://exports.feedly.com/ioc/
8a22cd92ac501da224308d248fd2e226/20220525.222341.all-ioc.md"
    }
]
}

```

```

    }
  ],
  "mentions": [
    {
      "text": "mail.saadzakhary[.]com:587",
      "type": "domain",
      "canonical": "mail[.]saadzakhary.com:587"
    },
    {
      "text": "hxxp://192.227.196[.]211/tea_shipping/
f_document_shp.doc",
      "type": "url",
      "canonical": "http://192[.]227.196.211/tea_shipping/
f_document_shp.doc"
    },
    {
      "text":
"f1794bfabeae40abc925a14f4e9158b92616269ed9bcf9aff95d1c19fa79352e",
      "type": "hash",
      "canonical":
"f1794bfabeae40abc925a14f4e9158b92616269ed9bcf9aff95d1c19fa79352e"
    }
  ]
}
}
],
"updated": 1643836615133
}

```

ThreatQuotient provides the following default mapping for this feed:

FEED DATA PATH	THREATQ ENTITY	THREATQ OBJECT TYPE OR ATTRIBUTE KEY	PUBLISHED DATE	EXAMPLES	NOTES
items[].title	Report.Value	N/A	items[].published	WordPress Contact Form Check Tester 1.0.2 XSS / Access Control	Title is stripped out from special characters and in in case it remains empty, it will be generated from: "Feedly Report title from: " + items[].published
items[].fullContent	Report.Description	N/A	items[].published	<!DOCTYPE html PUBLIC....	If fullContent does not exist, content.content is used, if that doesn't exist, summary.content is used.
items[].keywords	Report.Attribute/Tag	Tag	items[].published	My Software	Based on the Ingest Keywords user config will be saved as an Attribute, a Tag or both
items[].entities[].causes[]	Report.Attribute	Affected Software	items[].published	Windows 11	N/A
items[].origin.title	Report.Attribute	Origin	items[].published	Exploit Files \u2248 Packet Storm	N/A
items[].canonicalUrl	Report.Attribute	Source URL	items[].published	https://packetstormsecurity.com/files/165814/wpcfct102-xssaccess.txt	N/A
items[].summary.content	Report.Attribute	Feedly Summary	items[].published	WordPress Contact Form Check Tester plugin version 1.0.2 suffers from broken access control and cross site scripting vulnerabilities.	Chosen first over items[].leoSummary
items[].leoSummary[].sentences[].text	Report.Attribute	Feedly Leo Summary	items[].published	"Palo Alto Networks customers are protected against the..."	Only used if items[].summary doesn't exist
items[].categories[].label	Report.Attribute	Feedly Category	items[].published	Threat Intel	This attribute is the label of the feed it is associated with
items[].estimatedCVSS.category	Report.Attribute	Estimated CVSS Severity	items[].published	High	Estimated CVSS severity that Feedly has provided
items[].commonTopics[].label	Report.Attribute	Topic	items[].published	Cyber Security	Topics that Feedly categorized the item under
items[].themes	Report.Attribute	Common Subject	items[].published	Website	Topics that Feedly categorized the item under
items[].sources[].feedlyFeedType	Report.Attribute	Feedly Feed Type	items[].published	WebAlert	N/A
items[].entities[].label	Related.Indicator	CVE	items[].published	CVE-2022-0190	User chooses whether CVE's are ingested as Indicators or Vulnerabilities

FEED DATA PATH	THREATQ ENTITY	THREATQ OBJECT TYPE OR ATTRIBUTE KEY	PUBLISHED DATE	EXAMPLES	NOTES
items[].entities[].vulnerabilityInfo.cvssScore	Indicator .Attribute	CVSS Score	items[].published	5.4	N/A
items[].entities[].vulnerabilityInfo.hasExploit	Indicator .Attribute	Has Exploit	items[].published	True	N/A
items[].entities[].vulnerabilityInfo.hasPatch	Indicator .Attribute	Has Patch	items[].published	False	N/A
items[].entities[].label	Related.Vulnerability	N/A	items[].published	CVE-2022-0190	User chooses whether CVE's are ingested as Indicators or Vulnerabilities
items[].indicatorsOfCompromise.mentions[].canonical	Related.Indicator	FQDN	items[].published	bzone.no-ip.biz	N/A
items[].indicatorsOfCompromise.mentions[].canonical	Related.Indicator	URL	items[].published	https://sk5621.com.co	N/A
items[].indicatorsOfCompromise.mentions[].canonical	Related.Indicator	Email Address	items[].published	N/A	N/A
items[].indicatorsOfCompromise.mentions[].canonical	Related.Indicator	IP Address	items[].published	45.77.71.50:8082	N/A
items[].indicatorsOfCompromise.mentions[].canonical	Related.Indicator	MD5	items[].published	40b428899db353bb0ea244d95b5b82d9	N/A
items[].indicatorsOfCompromise.mentions[].canonical	Related.Indicator	SHA-1	items[].published	N/A	N/A
items[].indicatorsOfCompromise	Related.Indicator	SHA-256	items[].published	6fcd36052b242bc33e90577e9a9cf5dc91bc7c5f3ad587b0d45ab4a7cb7b73b3	N/A

FEED DATA PATH	THREATQ ENTITY	THREATQ OBJECT TYPE OR ATTRIBUTE KEY	PUBLISHED DATE	EXAMPLES	NOTES
.mentions[].canonical					
items[].indicatorsOfCompromise.mentions[].canonical	Related.Indicator	SHA-512	items[].published	40b428899db353bb0ea244d95b5b82d9	N/A
items[].entities[].label	Related.Attack Pattern / TTP	N/A	items[].published	T1187 - Forced Authentication	Depends on Ingest MITRE ATT&CK As user config (Attack Pattern or TTP).
items[].entities[].causes[]	Related.Identity	N/A	items[].published	Microsoft	N/A
items[].entities[].causes[]	Related.Malware	N/A	items[].published	TrickBot	Includes attribute of affected OS
items[].entities[].causes[]	Related.Adversary	N/A	items[].published	MuddyWater	N/A

Feedly Threat Intelligence

The Feedly Threat Intelligence feed ingests STIX 2.1 threat intelligence Reports from a configured Feedly stream. Depending on the selected configuration, the feed can also ingest related Indicators, Malware, Adversaries, and MITRE ATT&CK techniques as Attack Patterns or TTPs, providing comprehensive context for imported intelligence.

GET <https://cloud.feedly.com/v3/enterprise/ioc>

```
{
  "objects": [
    {
      "type": "report",
      "spec_version": "2.1",
      "id": "report--8d52f733-6826-4523-aad5-7f84b9f9a4df",
      "created": "2023-06-23T14:14:07.147182Z",
      "modified": "2023-06-23T14:14:07.147182Z",
      "name": "Bluepurple Pulse: week ending June 25th",
      "description": "<div>[Redacted]</div>",
      "published": "2023-06-23T07:13:58.781Z",
      "object_refs": [
        "malware--52acea22-7d88-433c-99e6-8fef1657e3ad",
        "malware--8c9bcc7d-0484-4067-bc57-30f1036fbac4",
        "threat-actor--68391641-859f-4a9a-9a1e-3e5cf71ec376",
        "attack-pattern--d10cbd34-42e3-45c0-84d2-535a09849584"
      ],
      "external_references": [
        {
          "source_name": "Feedly article",
          "url": "https://feedly.com/i/entry/
T8Gn8hJy9MDXgPPEWPf3eKFiC22pQg9/
jkwNHgMRBDU=_188e89acb7d:3ae99fe:cee8d097"
        },
        {
          "source_name": "BinaryFirefly",
          "url": "https://bluepurple.binaryfirefly.com/p/bluepurple-
pulse-week-ending-june-64e"
        }
      ],
      "labels": ["Wordpress Vulns", "Threat Intel"]
    },
    {
      "type": "malware",
```

```

    "spec_version": "2.1",
    "id": "malware--52acea22-7d88-433c-99e6-8fef1657e3ad",
    "created": "2023-06-23T10:46:39.001644Z",
    "modified": "2023-06-23T10:46:39.001644Z",
    "name": "Chrysaor",
    "description": "With our partners and with technical support
from Amnesty Internationalâ€™s Security Lab, weâ€™ve been
investigating the use of the spyware called Pegasus and the Israeli
surveillance company, NSO Group, that sells it to foreign
governments.",
    "is_family": true,
    "aliases": ["JigglyPuff", "Pegasus"],
    "external_references": [
      {
        "source_name": "",
        "url": "https://android-developers.googleblog.com/2017/04/
an-investigation-of-chrysaor-malware-on.html"
      }
    ]
  },
  {
    "type": "malware",
    "spec_version": "2.1",
    "id": "malware--8c9bcc7d-0484-4067-bc57-30f1036fbac4",
    "created": "2023-06-23T10:46:36.671208Z",
    "modified": "2023-06-23T10:46:36.671208Z",
    "name": "BlackLotus",
    "description": "Cybersecurity researcher Scott Scheferman
reported that a new Windows UEFI rootkit, dubbed Black Lotus, is
advertised on underground criminal forums.",
    "is_family": true,
    "external_references": [
      {
        "source_name": "",
        "url": "https://kn0s-organization.gitbook.io/blacklotus-
analysis-stage2-bootkit-rootkit-stage/"
      }
    ]
  },
  {
    "type": "threat-actor",
    "spec_version": "2.1",

```

```

    "id": "threat-actor--68391641-859f-4a9a-9a1e-3e5cf71ec376",
    "created": "2023-06-23T10:46:39.228673Z",
    "modified": "2023-06-23T10:46:39.228673Z",
    "name": "Lazarus Group",
    "description": "Since 2009, HIDDEN COBRA actors have leveraged
their capabilities to target and compromise a range of victims; some
intrusions have resulted in the exfiltration of data while others
have been disruptive in nature. Commercial reporting has referred to
this activity as Lazarus Group and Guardians of Peace. Tools and
capabilities used by HIDDEN COBRA actors include DDoS botnets,
keyloggers, remote access tools (RATs), and wiper malware. Variants
of malware and tools used by HIDDEN COBRA actors include Destover,
Duuzer, and Hangman.",
    "aliases": [
        "WhoisHacking Team",
        "Lazarus",
        "Bluenoroff",
        "APT 38",
        "OperationTroy",
        "Hidden Cobra",
        "NICKEL GLADSTONE",
        "WhoisHackingTeam"
    ],
    "external_references": [
        {
            "source_name": "",
            "url": "https://threatpost.com/operation-blockbuster-
coalition-ties-destructive-attacks-to-lazarus-group/116422/"
        }
    ]
},
{
    "type": "attack-pattern",
    "spec_version": "2.1",
    "id": "attack-pattern--d10cbd34-42e3-45c0-84d2-535a09849584",
    "created_by_ref": "identity--c78cb6e5-0c4b-4611-8297-
d1b8b55e40b5",
    "created": "2020-01-17T16:10:58.592Z",
    "modified": "2022-04-21T16:13:00.598Z",
    "name": "Launch Agent",
    "description": "Adversaries may create or modify launch agents
to repeatedly execute malicious payloads as part of persistence. When

```

a user logs in, a per-user launchd process is started which loads the parameters for each launch-on-demand user agent from the property list (.plist) file found in `/System/Library/LaunchAgents/`, `/Library/LaunchAgents/`, and `~/Library/LaunchAgents/`. (Citation: AppleDocs Launch Agent Daemons) (Citation: OSX Keydnep malware) (Citation: Antiquated Mac Malware) Property list files use the `Label`, `ProgramArguments`, and `RunAtLoad` keys to identify the Launch Agent's name, executable location, and execution time. (Citation: OSX.Dok Malware) Launch Agents are often installed to perform updates to programs, launch user specified programs at login, or to conduct other developer tasks. \n\n Launch Agents can also be executed using the [Launchctl](https://attack.mitre.org/techniques/T1569/001) command. \n \nAdversaries may install a new Launch Agent that executes at login by placing a .plist file into the appropriate folders with the `RunAtLoad` or `KeepAlive` keys set to `true`. (Citation: Sofacy Komplex Trojan) (Citation: Methods of Mac Malware Persistence) The Launch Agent name may be disguised by using a name from the related operating system or benign software. Launch Agents are created with user level privileges and execute with user level permissions. (Citation: OSX Malware Detection)(Citation: OceanLotus for OS X) ",

```

    "kill_chain_phases": [
      {
        "kill_chain_name": "mitre-attack",
        "phase_name": "persistence"
      },
      {
        "kill_chain_name": "mitre-attack",
        "phase_name": "privilege-escalation"
      }
    ],
    "external_references": [
      {
        "source_name": "mitre-attack",
        "url": "https://attack.mitre.org/techniques/T1543/001",
        "external_id": "T1543.001"
      }
    ],
    "object_marking_refs": [
      "marking-definition--fa42a846-8d90-4e51-bc29-71d5b4802168"
    ],
    "x_mitre_attack_spec_version": "2.1.0",

```

```

"x_mitre_contributors": ["Antonio Piazza, @antman1p"],
"x_mitre_data_sources": [
  "Command: Command Execution",
  "File: File Creation",
  "File: File Modification",
  "Service: Service Creation",
  "Service: Service Modification"
],
"x_mitre_deprecated": false,
"x_mitre_detection": "Monitor Launch Agent creation through
additional plist files and utilities such as Objective-See's
KnockKnock application. Launch Agents also require files on disk for
persistence which can also be monitored via other file monitoring
applications.\n\nEnsure Launch Agent's <code> ProgramArguments </
code> key pointing to executables located in the <code>/tmp</code> or
<code>/shared</code> folders are in alignment with enterprise policy.
Ensure all Launch Agents with the <code>RunAtLoad</code> key set to
<code>true</code> are in alignment with policy. ",
"x_mitre_domains": ["enterprise-attack"],
"x_mitre_is_subtechnique": true,
"x_mitre_modified_by_ref": "identity--c78cb6e5-0c4b-4611-8297-
d1b8b55e40b5",
"x_mitre_permissions_required": ["Administrator", "User"],
"x_mitre_platforms": ["macOS"],
"x_mitre_version": "1.4"
}
],
"id": "bundle--78031701-ea33-4942-a189-aaa93fb3ad2c",
"type": "bundle"
}

```



The mapping for this feed is handled by the native ThreatQ STIX 2 parser - see the STIX 2.0 Data Mapping topic for more information. The value of the attribute `Modified At` is updated at ingestion.

Average Feed Run



Object counts and Feed runtime are supplied as generalities only - objects returned by a provider can differ based on credential configurations and Feed runtime may vary based on system resources and load.

Feedly

The following results were obtained with both **Indicators** and **Vulnerabilities** selected as the type of CVE to ingest and both **Tags** and **Attributes** selected as the type for keywords to ingest. There may be more or less objects depending on feed.

METRIC	RESULT
Run Time	< 1 minute
Adversary	1
Attack Pattern	27
Identity	5
Indicators	85
Indicator Attributes	20
Malware	3
Report	13
Report Attributes	69
Vulnerability	6

Feedly Threat Intelligence

METRIC	RESULT
Run Time	1 minute
Adversary	100
Adversary Attributes	4,188
Attack Pattern	76
Attack Pattern Attributes	1,516
Indicators	279
Indicator Attributes	573
Malware	30
Malware Attributes	603
Report	20
Report Attributes	99

Known Issues / Limitations

- You cannot pull from a Personal Feed. You can only pull from streams/feeds that are listed under your Team Feeds.
- Feedly Cloud servers abuse-prevention systems may return 429 Too many requests for large volumes of data ingested.
- The **Feedly Threat Intelligence** feed might ingest data that contains characters not supported by the database, resulting in the feed finishing with a 500 Internal Server Error message. Special characters will be removed to prevent this error, and a permanent fix will be addressed in a future ThreatQ platform release.
- In regards to the **Parsed IOC Types** parameter - not all IOC types may be included in the Feedly API.

Change Log

- **Version 2.1.3**
 - Added a **MITRE ATT&CK Naming Format** configuration option to both Feedly feeds, allowing users to select the naming convention used for ingested MITRE ATT&CK techniques. This enhancement provides consistent ATT&CK technique naming and ingestion across the Feedly and Feedly Threat Intelligence feeds, whether techniques are imported as Attack Pattern or TTP objects.
- **Version 2.1.2**
 - Resolved an `UndefinedError` caused by missing alternate fields by implementing a safe fallback to an empty string.
 - Improved Feedly date handling by validating the `published` field and defaulting to `None` to prevent crashes from invalid or missing dates.
- **Version 2.1.1**
 - Added a new configuration parameter for both feeds:
 - **Ingest MITRE ATT&CK As** - select whether to ingest MITRE ATT&CK techniques as either Attack Patterns or TTPs.
- **Version 2.1.0**
 - Added the following new configuration parameter to all feeds:
 - **Parsed IOC Types** - select the IOC types to parse from the incoming content.
 - Added the following new configuration parameter to the **Feedly Threat Intelligence** feed:
 - **Do Not Ingest Description** - enable this parameter to exclude report descriptions from ingestion for the Feedly Threat Intelligence feed.
 - The **Ingest CVEs As** parameter for both feeds now allows multi-select.
 - Resolved a defect where API upload batches would fail if report titles included 4-byte Unicode characters.
- **Version 2.0.6**
 - Resolved an issue where special characters in the titles would cause the feed run to fail.
 - Added two new configuration parameters:
 - **Enable SSL Certificate Verification** - enable or disable verification of the server's SSL certificate.

- **Disable Proxies** - determine if the feed should honor proxy settings set in the ThreatQ UI.
- Added new Prerequisite entries for the integration:
 - All feeds require a valid API Token and Feedly API Stream.
 - The **Feedly Threat Intelligence** feed requires your Feedly account to have an Enterprise license.
- **Version 2.0.5**
 - Resolved an issue where report descriptions were not being populated if the CVEs were ingested as vulnerabilities.
 - The **Ingest CVEs As** parameter for both feeds will now only accept one selection: indicators or vulnerabilities.
- **Version 2.0.4**
 - Resolved a feed run issue caused by a missing MITRE Attack Technique identifier.
- **Version 2.0.3**
 - Resolved an issue with the Feedly Leo Summary that could cause feed run errors.
- **Version 2.0.2**
 - Added a client parameter, `client=theatq.integration`, to all API calls made by the integration. This change was made upon request from the provider.
 - Updated the default value for the **Ingest CVEs As** field. The default value is now set to Vulnerabilities.
- **Version 2.0.1**
 - Resolved an issue where users would encounter an error when reports did not contain a description.
- **Version 2.0.0**
 - Added new feed: **Feedly Threat Intelligence**.
 - Added improved Description formatting.
 - Updated minimum ThreatQ version to 5.6.0.
- **Version 1.1.0**
 - Added Published date to attributes ingested by the feed.
 - Added missing relationships.
 - Updated the default Indicator Status to **Review**.

- **Version 1.0.0**
 - Initial release